REMARKS

Reconsideration of the present application, as amended, is respectfully requested in view of the following remarks. Claims 1-8, 10-12, 14, 15, 17-19, 22, and 24 remain in the application after this Amendment is entered.

THE OFFICE ACTION:

Claims 1-8 and 17 stand rejected under 35 U.S.C. § 103(a) for obviousness over paragraphs 5-7 in the "Background of the Invention" of applicant's patent application in view of U.S. Patent No. 6,349,355 to Draves et al. (hereinafter Draves).

Claims 9-16 and 18-25 stand rejected under 35 U.S.C. § 102(e) for anticipation by Draves.

THE ART REJECTIONS:

Independent Claim 1

The Examiner rejected claims 1-8 and 17 for obviousness over paragraphs 5-7 in the "Background of the Invention" of applicant's patent application in combination with Draves. To establish a prima facie case of obviousness, the prior art references must teach or suggest all the claim limitations. MPEP § 2142.

Regarding claim 1, it is respectfully submitted a prima facie case for obviousness has not been established because none of the references disclose or suggest "a virtual memory mapping in the operating system address space" that allows a monitoring application to "read the performance data from the data structure without transferring the performance data to the user address space using a system interrupt." The Examiner relies on col. 4, line 66 to col. 5, line 5, and col. 3, lines 46-52, and Fig. 9 elements 52a and 52b of Draves for disclosing these claim limitations. However, none of these disclosures in Draves describe reading performance data stored in a data structure of an operating system address space from a user address space without using a system interrupt, as recited in claim 1.

For example, col. 4, line 66 to col. 5, line 5 of Draves describes a shared address range in both the user address space and the kernel address space that is designated for a shared executable component. The Draves disclosure does not mention reading performance data stored in a data structure of an operating system or kernel address space from a user address space without using a system interrupt, as claimed.

At the cited col. 3, lines 46-52, Draves describes that an "important benefit" of the kernel is its ability to access a calling process's virtual memory space to allow a user process to pass data addresses to kernel functions instead of passing the data. The Draves kernel can then use the addresses to "read, modify, and write memory in the user process' virtual address space." (Id.) Elements 52a and 52b in Fig. 9 merely identify position-dependent program modules in user shared system virtual address space and kernel shared system virtual address space. Nowhere does this disclosure teach or suggest reading performance data stored in a data structure of an operating system or kernel address space from a user address space without using a system interrupt, as claimed. Therefore, independent claim 1 is respectfully submitted to be patentable over the cited references at least for these reasons.

Furthermore, it is respectfully submitted that a prima facie case for obviousness has not been established because there is no suggestion or motivation in the cited references or in knowledge generally available to one of ordinary skill in the art to combine the "Background of the Invention" with Draves. The law is quite clear that for a claim to be held obvious, the prior art must suggest the modifications sought to be patented. In re Gordon, 221 USPQ 1125, 1127 (Fed. Cir. 1984). One must set aside in their minds that which the applicant teaches and follow the teachings from the references and what they fairly suggest. The Federal Circuit has cautioned that a person of ordinary skill is "one who thinks along the line of conventional wisdom in the art and is not one who undertakes to innovate." Standard Oil Co. v. American Cyanamid Co., 774 F.2d 448, 454 (Fed. Cir. 1985). It is impermissible simply to engage in a hindsight reconstruction of the claimed invention, using applicant's structure as a template and selecting references to fill the gaps. In re Gorman, 18 USPQ2d 1885.

The "Background of the Invention" discloses collecting sample data regarding the performance of a target application and transferring the data from the operating system to a user space using system calls and/or interrupts. Other than being directed to "virtual memory systems," Draves omits any disclosure directed to performance monitoring from a user address space that reads performance data within an operating system or kernel address space without the use of a system interrupt, as claimed. Since the "Background of the Invention" does not disclose or suggest any form of memory mapping or even the concept of memory mapping for performance monitoring (whether it be in the user address space or operating system address space), there is no motivation to combine the "Background of the Invention" with the virtual memory mapping of Draves--other than the impermissible use of hindsight reconstruction with claim 1 as a guide. Nevertheless, even if one did combine the two

references, all of the limitations of claim 1 would still not be met as described above. Therefore, it is respectfully submitted that claim 1 and claims dependent thereon (i.e., claims 2-8, 10-12, 14, 15, 17-19, and 22) are in condition for allowance.

Dependent Claims

Claim 2 is directed to the system of claim 1 including "one or more communication parameters" that are set to "pass information between the operating system and the monitoring application without a system call" and is patentable in its own right for at least this reason. The Examiner states that these claim limitations are disclosed in Fig. 9 of Draves. However, nothing in Fig. 9 discloses or fairly suggests that information is passed between the system virtual address space and the user virtual address space "without a system call." Moreover, Draves discloses the "when a user process calls a kernel function with a pointer argument, the pointer is biased before being dereferenced to account for the offset of the user address space within the kernel address space" (Abstract). This, as well as multiple references to a "system call handler" (see, e.g., col. 8, lines 24-28), suggests that transferring information between the user virtual address space and the system virtual address space in Draves is done using system calls. Based on the foregoing, it is respectfully submitted that claim 2 and claims dependent thereon (i.e., claims 3-5, 10-12, 14, and 15) are currently in condition for allowance.

Claim 3 is directed to the system of claim 2, further including "one or more parameters predefined to control handshaking between the operating system and the monitoring application" and is patentable in its own right for at least this reason. The Examiner states that the "handshaking" is taught by Draves at col. 7, line 55 to col. 8, line 18. However, this section in Draves describes a concept for virtual-to-physical memory mapping, such as is shown in Fig. 8, and use of an offset for mapping the user virtual address space to the system virtual address space, such as is shown in Fig. 7. Moreover, the term "handshaking" is not disclosed or fairly suggested in Draves. Notably, Draves does not disclose or fairly suggest use of "parameters predefined to control handshaking" as recited in claim 3. Therefore, based on the foregoing, it is respectfully submitted that claim 3 is currently in condition for allowance.

Claim 4 is directed to the system of claim 2, further including "a full flag that when set, instructs the monitoring application to read the performance data from the data structure" and is patentable in its own right for at least this reason. The Examiner states that the "reading" is taught by Draves at col. 7, line 55 to col. 8, line 18. However, this section in

Draves describes a concept for virtual-to-physical memory mapping, such as is shown in Fig. 8, and use of an offset for mapping the user virtual address space to the system virtual address space, such as is shown in Fig. 7. Moreover, the phrase "full flag" is not disclosed or fairly suggested in Draves. Notably, Draves does not disclose or fairly suggest use of a "full flag" as recited in claim 4. Therefore, based on the foregoing, it is respectfully submitted that claim 4 is currently in condition for allowance.

Claim 8 is directed to the system of claim 1, further including "a performance monitoring unit ... that collects the performance data of the one or more instructions and loads the performance data into the data structure" and is patentable in its own right for at least this reason. In rejecting claim 8, the Examiner states that the "performance monitoring unit" is taught by paragraphs 6 and 7 in the "Background of the Invention" of applicant's patent application. However, none of these paragraphs disclose or fairly suggest use of a "performance monitoring unit" or storing performance data associated with one or more instructions executing on the computer system as recited in claim 8. Paragraph 6, for example, merely discloses a monitoring application in a user space that is monitoring and evaluating the performance of a target application using system calls and/or interrupts to, for example, read sample data collected by the operating system regarding performance of the target application. The "Background of the Invention" does not describe the monitoring application as having the ability to "collect the performance data" and "load the performance data into the data structure," as recited in claim 8. Therefore, based on the foregoing, it is respectfully submitted that claim 8 is currently in condition for allowance.

As amended, claim 10 now depends from claims 1 and 2 and is directed to the system of claim 2, further including "header information" and is patentable in its own right for at least this reason. The Examiner states that "header information" is taught by Draves in Fig. 8. However, nothing in Fig. 8 discloses or fairly suggests "header information." Moreover, the phrase "header information" is not disclosed or fairly suggested in Draves. Therefore, based on the foregoing, it is respectfully submitted that claim 10 is currently in condition for allowance.

As amended, claim 11 now depends from claims 1 and 2. Claim 11 recites "a read parameter ... changeable by the operating system to indicate ... that the monitoring application may read the data stored in the data structure" and is patentable in its own right for at least this reason. The Examiner states that the "read parameter" is taught by Draves at col. 9, lines 11-19. However, this section of Draves merely discloses certain aspects of the copy-on-write page protection shown in Fig. 10. Draves discloses how the virtual memory system provides

copy-on-write page protection if either the user process or kernel writes to a page by copying the physical page frame to another location in memory, updating the virtual address space of the user process or the kernel to point to the copy, and then performing the write (col. 8, line 50 - col. 9, line 2). Col. 9, lines 11-19 goes on to describe how the user can pass arguments to the kernel which triggers the copy-on-write page protection. Notably, this section of Draves does not disclose or fairly suggest use of a read parameter or any type of read operation by a monitoring application, as recited in claim 11. Therefore, based on the foregoing, it is respectfully submitted that claim 11 is currently in condition for allowance.

As amended, claim 22 now depends from claim 1 and is directed to the system of claim 1, further including "an overflow data structure" and is patentable in its own right for at least this reason. The Examiner states that Draves teaches an "overflow data structure" when the data structure is full at col. 9, lines 45-65. However, this section of Draves discloses certain aspects of offsetting user virtual memory within the kernel address space as shown in Fig. 11. In the example, Draves discloses how the user virtual memory is offset by 40000000h in the kernel address space (col. 9, lines 45-65). Notably, this section of Draves does not disclose or fairly suggest use of an overflow data structure as recited in claim 22. Moreover, the term "overflow" is not disclosed or fairly suggested in Draves. Therefore, based on the foregoing, it is respectfully submitted that claim 22 is currently in condition for allowance.

Independent Claim 24

As amended, independent claim 24 now essentially incorporates the limitations previously presented in dependent claim 25. More specifically, claim 24 now recites "one or more control parameters associated with the data structure where communication handshaking is established between the kernel address space and the user address space by setting predetermined values for the one or more control parameters." The Examiner states that the "handshaking" is taught by Draves at col. 4, line 60 to col. 5, line 23 and col. 7, lines 16-41. However, these sections in Draves describe a concept for shared virtual memory, such as is shown in Figs. 7 and 8. Moreover, the term "handshaking" is not disclosed or fairly suggested in Draves. Notably, Draves does not disclose or fairly suggest use of "control parameters associated with the data structure" for "communication handshaking" as recited in claim 24. Therefore, based on the foregoing, it is respectfully submitted that claim 24 is currently in condition for allowance.

CONCLUSION

Based on the foregoing amendments and remarks, the applicant believes that all of the claims in this case (i.e., claims 1-8, 10-12, 14, 15, 17-19, 22, and 24) are now in a condition for allowance and an indication to that effect is earnestly solicited. Furthermore, if the Examiner believes that additional discussions or information might advance the prosecution of this case, the Examiner should feel free to contact the undersigned at the telephone number indicated below.

Respectfully submitted,

Nenad Pejic (Reg. No. 37,415)
Calfee, Halter & Griswold, LLP
(216) 622-8835